UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/998,756	12/03/2001	Jeff L. Hunter	TI-33109	6454

23494 7590 10/05/2004

TEXAS INSTRUMENTS INCORPORATED
P O BOX 655474, M/S 3999
DALLAS, TX 75265

EXAMINER	
RUTTEN, JAMES D	

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 10/05/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/998,756	HUNTER ET AL.	
	Examiner	Art Unit	
	J. Derek Rutten	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *03 December 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *03 December 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☒ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-24 have been examined.

Oath/Declaration

2. The oath or declaration is defective. A new oath or declaration in compliance with 37 CFR 1.67(a) identifying this application by application number and filing date is required. See MPEP §§ 602.01 and 602.02.

> The oath or declaration is defective because:
> While acknowledging a duty to disclose as defined in 37 CFR 1.56(a), it does not state that the person making the oath or declaration acknowledges the duty to disclose to the Office all information known to the person to be material to patentability as defined in 37 CFR 1.56 in its entirety, as required by 37 CFR 1.63(b)(3).

Double Patenting

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4. Claims 1-12, 17-20, and 21-24 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-4, 6, and 7 of copending Application No. 09/998,755. Although the conflicting claims are not identical, they

are not patentably distinct from each other because they are obvious variations that recite a technique for maintaining coherency in a multiprocessor debugging environment.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

5. Claims 1-24 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-5, 7-12, and 16-18 of copending Application No. 09/998,329. Although the conflicting claims are not identical, they are not patentably distinct from each other because they are obvious variations that recite a technique for maintaining coherency in a multiprocessor debugging environment.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

6. Claims 1-12, 17-20, and 21-24 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-5, 7-12, and 16-18 of copending Application No. 09/998,330. Although the conflicting claims are not identical, they are not patentably distinct from each other because they are obvious variations that recite a technique for maintaining coherency in a multiprocessor debugging environment. While the '330 application does not expressly disclose setting or clearing breakpoints, "How Debuggers Work" by Rosenburg teaches a well known method for setting and clearing breakpoints (pages 40-41) by saving and replacing a location in memory with a special instruction to set a breakpoint, then replacing the special instruction with the saved memory contents to clear the breakpoint. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Rosenburg's breakpoints with the multiprocessor

debugger of the '330 application. One of ordinary skill would have been motivated to stop execution of a program in order to examine state and ensure proper execution and/or fix bugs.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. Claim 14 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

9. Claim 14 recites the limitation "the software memory map" in line 2. There is insufficient antecedent basis for this limitation in the claim. This limitation will be interpreted as --a software memory map--.

Claim Rejections - 35 USC § 102

10. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

11. Claims 1, 17, and 21 are rejected under 35 U.S.C. 102(a) as being anticipated by U.S. Patent 6,065,078 to Falik et al. (hereinafter "Falik").

As per claim 1, Falik discloses:

A method for maintaining coherency of software breakpoints in shared memory when debugging a multiple processor system (Falik column 17 lines 28-67: "Synchronization is achieved by means of shared memory or shared semaphore."), *the method comprising the steps of:*

activating a first debug session associated with a first processor of a plurality of processors and at least a second debug session associated with a second processor of the plurality of processors (Falik column 2 lines 37-43: "FIG. 18 illustrates the overall architecture of a system 1800 including a multiprocessor integrated circuit (Normandy 1810) and a host computer (HOST 1820) having a debugger (1830a to 1830c) for each processor (1840a to 1840c) that interacts in a debugger interface module 1841, with a separate monitor executing on each separate processor.");

setting a first software breakpoint in a shared memory location in the first debug session such that all debug sessions are notified of the setting of the breakpoint (Falik column 5 lines 17-21: "The ABORT may be triggered for a processor for which a corresponding abort mask bit in the ABORT mask register of the TAP 102 is set. Then, either a "SCAN TX", with PID=Il 12, or a write of 1 to the respective bit in the DBGABORT register will cause the abort.") *and*

clearing the first software breakpoint in the shared memory location in the second debug session such that all debug sessions are notified of the clearing of the breakpoint (Falik column 7 lines 49-50: " `Each ISE interrupt is cleared when DBGISESRC.ABORT.sub_i and IDBGISESRC.RX._i both equal 0.`").

As per claim 17, Falik discloses:

A software development system (column 20 line 2 – column 22 line 7), *comprising:*

a memory storage system holding a software development tool program (Fig. 21)*;*

a host computer connected to the memory storage system, the host computer operable to execute the software development tool program (Fig. 18)*;*

a test port for connecting to a target hardware system, the hardware system being comprised of multiple processors with common shared memory and operable to execute an application program (Fig. 19 and Fig. 21).

All further limitations have been addressed in the above rejection of claim 1.

As per claim 21, Falik discloses:

A digital system (column 20 line 2 – column 22 line 7), *comprising:*

multiple processors with common shared memory for executing an application program (FIG. 21)*; and*

All further limitations have been addressed in the above rejection of claim 1.

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. Claims 2-7, 18-20, and 22-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Falik as applied to claims 1, 17, and 21 above, and further in view of U.S. Patent 6,088,770 to Tarui et al. (hereinafter "Tarui").

As per claim 2, the above rejection of claim 1 is incorporated. Falik does not expressly disclose a memory map. However, in an analogous environment, Tarui teaches: *the step of creating a software memory map of the memory usage of a plurality of processors in the system to be debugged* (Tarui column 9 lines 12-31). It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Tarui's method of using a memory map with Falik's multiple processors. One of ordinary skill would have been motivated to track which processors had control of which ranges of memory so that other processors would be restricted from modifying the contents of that range.

As per claim 3, the above rejection of claim 2 is incorporated. Falik further discloses:

updating a software representation maintained for software breakpoints for each of the first plurality of processors (Falik column 16 lines 36-45); *and*

writing the software breakpoint instruction in the shared memory location (Falik column 16 lines 36-45).

Falik does not expressly disclose searching a software memory map. However, Tarui teaches: *searching the software memory map to find a first plurality of processors having read access to the shared memory location* (Tarui column 6 line 66 – column 7 line 9).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Tarui's method of searching a software memory with Falik's debugger. One of ordinary skill would have been motivated to maintain synchronized data in a multiprocessor computer. Data maintenance ensures that correct code is being executed.

As per claim 4, the above rejection of claim 3 is incorporated. Falik does not expressly disclose selecting a processor to execute. However, Tarui teaches: *wherein the step of writing comprises a method for selecting a processor to execute the write, the method comprising the steps of:*

if the first processor associated with the first debug session requesting the setting of the software breakpoint has write access to the shared memory location then selecting the first processor to perform the write request (Tarui column 9 lines 42-47);

else performing the following steps a-b:

a. searching the software memory map for a second processor with write access to the shared memory location (Tarui column 9 lines 32-41)*;*

b. selecting the second processor to perform the write request (Tarui column 15 lines 46-52)*; and*

passing the software breakpoint instruction to the selected processor to be written into the shared memory location (Tarui column 14 lines 22-26; also column 15 lines 46-52).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Tarui's method for selecting a processor with Falik's multiprocessor debugger. One of ordinary skill would have been motivated to reduce congestion and access latency when debugging software executing on a plurality of processors.

As per claim 5, the above rejection of claim 4 is incorporated. Falik does not expressly disclose passing a software breakpoint instruction. However, Tarui teaches: *wherein the step of passing the software breakpoint instruction comprises the steps of:*

searching the software memory map for a second plurality of processors that have read access to the shared memory location (Tarui column 6 line 66 – column 7 line 9)*;*

broadcasting the write request to the second plurality of processors (Tarui column 10 line 66 – column 11 line 5)*; and*

performing cache coherency updates in response to the write request in each of the second plurality of processors as required (Tarui column 6 lines 46-53; also column 3 lines 5-11).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Tarui's method of passing a breakpoint with Falik's debugger. One of ordinary skill would have been motivated to maintain a coherent representation of a breakpoint as with other data. A coherent representation of a breakpoint ensures that all processors halt at the same step of a program.

As per claim 6, the above rejection of claim 5 is incorporated. Falik does not expressly disclose broadcasting the write request. However, Tarui teaches: *wherein the step of broadcasting the write request comprises indicating that the write request is intended for maintaining cache coherency as opposed to a normal write request* (Tarui column 14 lines 5-14; also column 8 lines 7-11). All further limitations have been addressed in the above rejection of claim 5.

As per claim 7, the above rejection of claim 6 is incorporated. Falik does not expressly disclose performing cache coherency updates. However, Tarui teaches: *wherein the step of performing comprises using cache coherency capabilities, if any, of a processor in response to the write request intended for maintaining cache coherency* (Tarui column 8 lines 7-11). All further limitations have been addressed in the above rejection of claim 5.

As per claims 18-20, the above rejection of claim 17 is incorporated. All further limitations have been addressed in the above rejection of claims 2-4, respectively.

As per claims 22-24, the above rejection of claim 21 is incorporated. All further limitations have been addressed in the above rejection of claims 2-4, respectively.

14. Claims 8-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Falik and Tarui as applied to claim 2 above, and further in view of U.S. Patent 6,708,326 to Bhattacarya.

As per claim 8, Falik does not expressly disclose: *writing the original instruction stored in a software representation maintained for software breakpoints into the shared memory location.* However, in an analogous environment, Bhattacarya teaches the well known method of clearing a breakpoint by restoring the original instruction back to the memory location where the breakpoint formerly resided (column 2 lines 6-17).

All further limitations have been addressed in the above rejection of claim 3.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Bhattacarya's teaching of breakpoints with Falik's debugger. One of ordinary skill would have been motivated to continue normal execution after encountering a breakpoint. The integrity of a program is restored and normal execution is confirmed by replacing the original instruction.

As per claim 9-12, the above rejection of claim 8 is incorporated. All further limitations have been addressed in the above rejection of claims 4-7, respectively.

15. Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over Falik as applied to claim 1 above, and further in view of "How Debuggers Work" by Rosenberg.

As per claim 13, the above rejection of claim 1 is incorporated. Falik does not expressly disclose a method of stepping over code. However, in an analogous environment, Rosenberg teaches the well known method of stepping over source code:

requesting that a software breakpoint in a shared memory location be stepped over or program execution resumed after hitting the breakpoint in a third debug session (This is inherent in debugging since the alternative is terminating the program. Further, see Figure 6.3.);

clearing the software breakpoint in the shared memory location in the third debug session such that all debug sessions are notified of the clearing of the breakpoint (bottom of page 41);

stepping a processor associated with the third debug session to the instruction after the shared memory location from which the software breakpoint was cleared (bottom of page 41); *and*

setting the first software breakpoint in the shared memory location in the third debug session such that all debug sessions are notified of the setting of the breakpoint (bottom of page 41).

Rosenberg further discusses synchronization and notification of processors by a debugger on page 203 under "Multiprocessor Breakpoint Issues".

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Rosenberg's teaching of single-stepping with Falik's debugger. One of ordinary skill would have been motivated to ease into a problem area of code in order to understand and evaluate the incrementally evolving state of the program. This can be accomplished with the single-step mechanism taught by Rosenberg.

16. Claims 14-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Falik and Rosenberg as applied to claim 13 above, and further in view of Tarui.

As per claim 14, the above rejection of claim 13 is incorporated. Falik discloses halting (column 5 lines 5-8). Falik does not expressly disclose searching a software memory map. However, Tarui teaches searching a memory map for processors having read access to shared memory (Tarui column 6 line 66 – column 7 line 9). Further, Rosenberg teaches halting after requesting and before clearing (page 41). It would have been obvious to one of ordinary skill in the art at the time the invention was made to use Rosenberg's teaching of halting and Tarui's teaching of a memory map with Falik's debugger. One of ordinary skill would have been motivated to synchronize the execution of parallel processors during a halt condition in order to examine a snapshot of the state of a running process.

As per claims 15 and 16, the above rejection of claim 13 is incorporated. All further limitations have been addressed in the above rejection of claims 3 and 8, respectively.

Conclusion

17. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. "Code Composer User's Guide" by Texas Instruments, published February 1999, substantially teaches many of the claim limitations including a debugger that can set and clear breakpoints in a multiprocessor system (See page 2-4 Section 2.1.2 and page 3-9 Section 3.7). This reference also teaches using a memory map and single-stepping code (See page 2-12 Section 2.5 and Chapter 7), but was not yet relied upon.

18. Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (703) 605-5233. The examiner can normally be reached on M-F 6:30-3:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

After October 25, 2004, examiner can be reached at new telephone number (571) 272-3703, and the examiner's supervisor, Tuan Q. Dam can be reached at (571) 272-3694.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr

TUAN DAM
SUPERVISORY PATENT EXAMINER